

In the Abstract:

Please replace the Abstract of the Disclosure with the new Abstract of the Disclosure, attached hereto as a separate sheet.

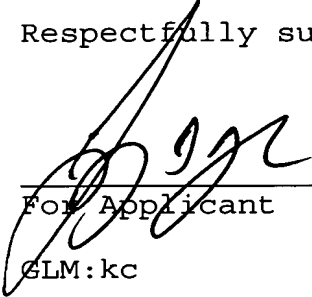
Remarks:

This preliminary amendment is being submitted in an effort to correct informalities and provide improved translations of certain wording in the initial application. The changes are not made for any reason related to the statutory requirements for a patent nor do they narrow the scope of the claim for any reason related to the statutory requirements for a patent. The newly entered claims are fully supported in the original claims.

An early action on the merits of the application is solicited.

Please charge any fees that might be due with respect to  
Sections 1.16 and 1.17 to the Deposit Account of Lerner and  
Greenberg, P.A., No. 12-1099.

Respectfully submitted,



---

For Applicant

GLM:kc

June 15, 2001

Lerner and Greenberg, P.A.  
P.O. Box 2480  
Hollywood, Florida 33022-2480  
Tel.: (954) 925-1100  
Fax: (954) 925-1101



Page 1 of Specification With Markings to Show Changes Made:

Page 1, replace the paragraph beginning on line 54, with:

--Summary of the Invention:

The [present] invention is thus based on the object of providing a translation system which can be used universally, whose construction is less complex, and which produces verifiable results. [According to the invention, this object is achieved by the] The invention includes a method for converting interface definitions within source code programs into an intermediate format [as claimed in independent patent claim 1] by a computer system that carries out the method, including the steps of identification of at least one object in the source code program, and identification of at least one interface in the at least one identified object. At least one of the identified interfaces may be an internal interface for producing a link from objects within the source: code program and/or at least one of the identified interfaces may be an external interface for producing a link from an object with interfaces located outside the source code program. The at least one interface may be an input interface and/or an output interface. The method further includes identification of at least one internal link between at least one output interface and at least one input interface between at least two objects and/or identification of at least one external link of the at least one external interface. An at least two dimensional

intermediate format table having rows arranged in a first dimension, having rows arranged in a second dimension, and having cells at the intersections of the first and second rows is created. The rows in the first dimension are assigned designations for each of the at least one identified objects. The rows in the second dimension are assigned designations for each of the at least one identified links. Designations for the output interface and/or input interface that is associated with both the respective identified object and the identified internal link are inserted in each of those cells that are located at the intersection of one of the rows in the first dimension with the designation of an identified object and one of the rows in the second dimension with the designation of an identified internal link, and/or the designations for the output interface and/or input interface that is associated with both the respective identified object and the identified external link are inserted in each of those cells that are located at the intersection of one of the rows in the first dimension with the designation of an identified object and one of the rows in the second dimension with the designation of an identified external link.

[by] The invention further includes a method for converting interface definitions from an at least two-dimensional intermediate format table into object program code [according to the independent patent claim 17, and by] the intermediate

format table having first rows arranged in a first dimension,  
second rows arranged in a second dimension and cells at the  
intersections of the first and second rows. Rows in the first  
dimension are assigned designations for at least one object.  
Rows in the second dimension are assigned designations for at  
least one internal link between the objects and/or at least  
one external link of an object. Designations for an internal  
output interface and/or internal input interface that is/are  
associated with both the respective object and the link are  
inserted in each of those cells that are located at the  
intersection of one of the rows in the first dimension with  
the designation of an object and one of the rows in the second  
dimension with the designation of an internal link, and/or in  
which designations for the external output interface and/or  
external input interface that is/are associated with both the  
respective object and the external link are inserted in each  
of those cells that are located at the intersection of one of  
the rows in the first dimension with the designation of an  
object and one of the rows in the second dimension with the  
designation of an external link. A computer system carries  
out the method with the steps of creating at least one program  
code object based on information, contained in the  
intermediate format table, about the at least one object.  
Associated internal output interfaces and/or internal input  
interfaces are assigned to their program code object. At least  
one link between program code objects is created based on the

information, contained in the intermediate format table, about the internal links of the internal input interfaces and internal output interfaces; and/or associated external output interfaces and/or external input interfaces are assigned to their program code objects.

The invention also includes an intermediate format table which is suitable for converting interface definitions from an intermediate format table into object program code. [this purpose, according to the independent patent claim 25] The intermediate format table for storing interface information, which is contained in program code, in a computer system has at least two dimensions, rows arranged in a first dimension, rows arranged in a second dimension, and cells at the intersections of the first and second rows. Rows in the first dimension are assigned designations for at least one object in the program code. Rows in the second dimension are assigned designations for at least one internal link between objects and/or designations for at least one external link of the program code. Designations for an output interface and/or input interface that is/are associated with both the respective object and the internal link are inserted in each of those cells that are located at the intersection of one of the rows in the first dimension with the designation of an object and one of the rows in the second dimension with the

designation of an internal link, and/or designations for the output interface and/or input interface that is/are associated with both the respective object and the external link are inserted in each of those cells that are located at the intersection of one of the rows in the first dimension with the designation of an object and one of the rows in the second dimension with the designation of an external link.--.

Page 19, replace the paragraphs in lines 20 to 31 with:

--[In the following text, two examples will be used to illustrate how program code can be converted into an intermediate format table. In this case, reference is made to the drawings, in which:]

Brief Description of the Drawings:

[Figure] FIG. 1 is [shows] an embodiment of [the] an intermediate format table [of the present invention which has been] produced from a first exemplary program according to the invention; and

[Figure] FIG. 2 is [shows] a further embodiment of the intermediate format table of [the present invention, which has been] FIG. 1 produced from a second exemplary program according to the invention.

Description of the Preferred Embodiment:

In all the figures of the drawing, sub-features and integral parts that correspond to one another bear the same reference symbol in each case.

In the following text, two examples will be used to illustrate how program code can be converted into an intermediate format table, with reference made to the drawings.--.